Shock-conforming Mesh Generation for Aerodynamic Analyses at Supersonic Regimes

J.Herrera-Montojo*, M. Fossati, E. Minisci

Aerospace Centre of Excellence
University of Strathclyde

75 Montrose street, Glasgow, United Kingdom

Abstract

A shock estimation approach is proposed in the context of the generation of shock-conforming meshes for the numerical analysis of inviscid, steady, supersonic and hypersonic flows. For given flow conditions and vehicle's geometry, the method provides a fast estimation of the shock waves pattern such that grid points can be clustered along the shock waves in a judicious manner. In this way, the uncertainty on mesh generation for shock-dominated flows is reduced and the use of adaptive mesh refinement could be made more efficient or, in some cases, even considered not necessary. The approach is verified against two- and three-dimensional supersonic flows for conceptual exemplary geometries like wedges and revolution bodies and more real-world vehicles configurations like rockets and hypersonic aircraft. Qualitative and quantitative assessment of the solution-mesh pair quality is proposed to evaluate the quality of the resulting shock-conforming meshes.

Keywords: Supersonic aerodynamics, Shock estimation, Mesh generation.

*Corresponding author
Email address: javier.herrera@strath.ac.uk (J.Herrera-Montojo)

1. Introduction

Computational Fluid Dynamics (CFD) has established itself as a powerful tool to support, together with experimental analysis and flight testing, the design of future air vehicles. The extensive adoption of CFD throughout the entire design process is nevertheless still limited by the high computational resources that are often required to solve complex 3D flows over elaborated vehicle's geometric configurations. A lot of efforts are therefore continuously put forward by researchers and engineers to streamline and certify as much as possible the CFD process in order to reduce uncertainty and improve the design. This task is particularly challenging in the case of vehicles that are deemed to operate at supersonic and hypersonic regimes where it becomes important to accurately address the complex pattern of shock waves and the resulting pressure, shear stress and heat loads distribution along the aircraft's surface [1, 2].

In order to capture accurately the strong gradients and the flow physics associated to high-Mach regimes, it is essential to use appropriate mesh spacing and alignment near the shock waves such that the discretization error of the numerical solution is either minimized or equidistributed over the entire domain [3]. The position of the shocks, however, is usually only known after the simulation is completed and appropriate techniques have to be adopted to minimize the mesh-related numerical diffusion and to promote grid independence of the numerical results [4].

A commonly adopted approach is to heuristically rely on the engineering sense in estimating the shocks positioning, even if this method can not consistently provide any guarantee that the resulting mesh would be optimal (according to one or more existing mathematical criteria). Alternatively, the shock structure could be estimated through analytical approaches, but the existing ones are limited to very specific geometries and flow conditions[5]. Examples of

this are the work of Catalano[6], defining the far field boundary of a CFD simulation of the Vega launcher using Billig's[7] approximate solution for bow shock shape and stand off distance for spheres and cylinders in supersonic flow; or Carter[8], applying mesh alignment and refinement along the free stream Mach angle for sonic boom signature predictions.

A more rigorous and consistent approach to get appropriate shock-conforming meshes is represented by mesh adaptation techniques that iteratively improve the positioning of the nodes and the elements of the mesh on the basis of the solutions obtained at previous iterations of the process. Central to any mesh adaptation technique is the definition of suitable error estimators and the identification of methods to adapt the elements of the mesh to fit at best (in a certain mathematical sense depending on the method used) the flow field under investigation. The resulting method allows rigorous and quantitative control on the accuracy of the solution but has the drawback of being often a lengthy and time-consuming approach.

The computational effort of mesh adaptation could be relieved if some sort of a-priori consistent information can be used before or together with the available CFD solution. A recent interesting contribution to the field of mesh adaptation is the one from Gauci and collaborators [9], who proposed an a-priori error-based mesh adaptation technique in the attempt of mitigating the necessity of dealing always with a-posteriori error estimation. Remaki and collaborators[10] also formulated an interesting approach introducing in the process a-priori approximations of the flow field obtained by a technique based on some simplifications on the physics. In their work, they were able to provide an optimized initial mesh for a viscous supersonic flow by using information coming from a preliminary inviscid simulation. Differently from the aforementioned approximated methods, this approach is capable of dealing with any type of geometrical con-

figuration, but suffers from the cost of a full inviscid CFD simulation, which in some cases might become quite time consuming, especially when dealing with complex three-dimensional vehicle configurations.

The present work aims at formulating a cohesive and optimized approach capable of providing fairly accurate estimates of the inviscid and steady shock waves pattern around supersonic vehicles without the necessity to run any expensive CFD simulation and such that it would become possible to obtain initial shock-conforming meshes also in the case of complex three-dimensional vehicle configurations. The aim is to propose a technique that could be considered halfway between existing methods for very simple geometries and approaches that require the numerical solution of some sort of simplified system of equations (i.e. Euler, or panel methods). In this sense, techniques for shock estimation already available in the literature will be reconsidered and recast in a novel and optimized fashion.

The manuscript is structured as follows: In Section 2 the shock estimation methodology will presented and the relevant theoretical background will be illustrated. Section 3 will illustrate a series of verification cases used to establish quantitatively the accuracy of the method over conceptual geometries. Section 4 will present an error estimator suitable for mesh comparison. Eventually, Section 5 will show the application of the proposed formulation to the generation of shock-conforming meshes for three-dimensional vehicles, namely a rocket and the X43 hypersonic vehicle. In this section, an analysis of the resulting accuracy of the solution will be presented by qualitative and quantitative comparison of the shock-conforming mesh-solutions to the solutions obtained on uniform meshes.

2. Shock Estimation

During the early days of supersonic flight theory there was an abundance of approximate models that provided the shock location, either for its own sake or as a by-product of the computation of pressures over the surface of the vehicle[11]. The interest for these approximations can be explained by the need of predictive capability for the design of supersonic vehicles coupled with very limited computational resources. These methods were implemented in low order aero-thermal models like the supersonic/hypersonic arbitrary body program (S/HABP)[12] for preliminary design of supersonic and hypersonic vehicles.

The proposed procedure expands on classical methods, addressing two and three dimensional geometries with attached and detached shocks as well as shock structures composed of multiple shocks in one cohesive methodology. Specifically, the treatment for detached shocks with subsonic zone approximation patched to the attached shock estimation method under a correction factor, the engineering formula for detached shocks around general 3D stagnation points and the use of a clustering algorithm for multi-shock reconstruction are all original contributions.

The shock estimation procedure presented here has its roots in the tangent wedge method and shock expansion theory, two classical tools for approximate calculations of surface pressures on a body in supersonic flight[13].

The tangent wedge method equates the pressure at any point of the surface of the vehicle with the one on the surface of a wedge with the same inclination with respect to the flow as the local inclination at the point and at the same free stream conditions. The pressure for the equivalent wedge is then computed through the use of the oblique shock relations. An experimental correlation can be used to extend its application to points where the equivalent wedge produces

a detached shock. This procedure has very weak theoretical foundations. It is used because it produces reasonable results with very little effort in a wide range of flow conditions, but it lacks a strong physical justification[13].

Shock expansion theory is used in convex bodies with attached shocks. Here, the geometry is discretized into segments. The conditions at the first one are computed as the after shock conditions of the attached oblique shock. The conditions on subsequent segments are computed through a Prandtl-Meyer expansion fan between them, advancing along the surface of the body. In its first order approximation, the method produces constant values inside each of the discretized segments of the geometry. In contrast with tangent wedge method, shock expansion theory can be explained from a physical perspective. It can be seen as a simplification of the method of characteristics for two dimensional, steady, isentropic, irrotational flows, ignoring the effect of the right running family of characteristics. That is, ignoring the effect of the reflected perturbations on the shock over the body. This assumption is closer to reality as the Mach number of the free stream increases and as the body becomes more slender[13].

The description of the present method will follow a constructive approach. It is first developed for highly particular bodies (two dimensional convex geometries with attached shocks in inviscid flow) and then generalized, introducing modifications that allow for the removal of assumptions on the geometry.

2.1. Shock estimation in two dimensions

For attached two-dimensional shocks, and considering the geometry to be a polyline, the shock estimation procedure is as follows.

- The shock for the first segment of the geometry is computed as an attached wedge shock. Its inclination given by solving for shock inclination (β) in

Eq. 1 through a numerical root finding algorithm.

$$\tan\theta = 2\cot\beta\frac{M_1^2\sin^2\beta - 1}{M_1^2(\gamma + \cos 2\beta) + 2} \tag{1}$$

θ is the inclination of the wedge, M_1 the Mach number before the shock and γ the ratio of specific heats.

- The next shock section has the inclination given by the equivalent wedge of the second segment of the polyline. Its origin is determined by the intersection of the first shock segment with the first Mach wave of the expansion fan produced at the end of the first segment of the geometry. The inclination of this Mach wave is given by $\mu = \arcsin(1/M_2)$, where the M_2 is the after shock Mach number, computed through the oblique shock relations

$$M_2 = \frac{1}{\sin(\beta-\theta)}\sqrt{\frac{1 + \frac{\gamma-1}{2}M_1^2\sin^2\beta}{\gamma M_1^2\sin^2\beta - \frac{\gamma-1}{2}}} \tag{2}$$

- This continues for all sections of the geometry polyline.

The procedure, illustrated in Fig. 1, is an advancing front method, where the errors are cumulative, as the position of each local shock is influenced by the position of the previous one. Fig. 2 shows the shock estimation against CFD results. There is good agreement between the two solutions. The cumulative error in the streamwise direction can also be seen in the way the estimated shock slowly drifts away from the CFD solution.

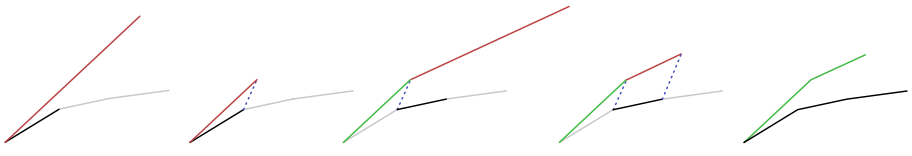

Figure 1: Procedure for shock estimation.

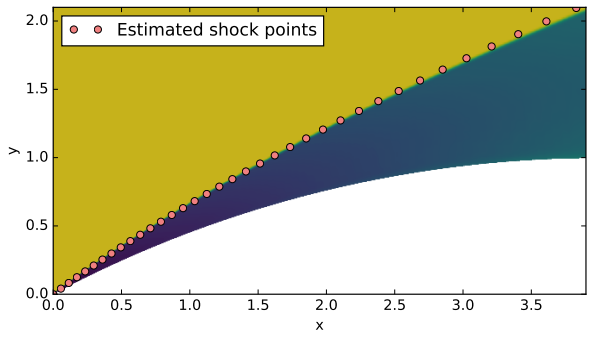

Figure 2: Shock estimation for an ogive over CFD Mach field

The resulting method considers the expansion between segments as infinitesimal Mach waves, which is reasonable if the polyline is a discretization of a C^1 underlying geometry, and takes the local shock inclination and Mach number from the tangent wedge method.

2.1.1. Detached shocks

For detached shocks, the starting procedure employed on the attached shock case is not applicable. There is no attached equivalent wedge in the first segment of the geometry. A solution for the detached zone must be imposed. Assuming the geometry to be a circle in the vicinity of the stagnation point, Billig [7] provides such a solution. The detached shock around the stagnation point is described as a hyperbola

$$x = R + \Delta - R_c \cot^2\theta \left[\left(1 + \frac{y^2 \tan^2\theta}{R_c^2} \right)^{1/2} \right] \tag{3}$$

being R the radius of curvature of the geometry at the stagnation point, R_c the radius of curvature of the shock at the vertex, Δ the stand off distance and θ the asymptotic angle of the hyperbola. This is the free stream Mach angle in

the case of a cylinder and the attached shock angle in the case of a cylinder with a wedge afterbody. The axes have their origin at the centre of curvature of the geometry, with the x axis aligned with the flow. The vertex radius of curvature and stand off distance are given by

$$R_c/R = 1.386 \ \exp[1.8/(M-1)^{0.75}] \tag{4}$$

$$\Delta/R = 0.386 \ \exp(4.67/M^2) \tag{5}$$

where M is the Mach number of the free stream. The shock estimation is patched to this hyperbola at the inclination where the first equivalent wedge from the geometry produces an attached weak shock (post shock Mach number is supersonic). The estimation process then continues as in the attached case.

While this procedure provides a suitable start for the shock estimation method, the results are far worse than the ones obtained in the case of attached shocks. This is not surprising, though, as the assumptions made earlier do not hold any more. The flow after the bow shock is rotational and its evolution after is non isentropic.

A careful look at the estimation of the shock for a cylinder suggest a way to improve the results. In the procedure presented here, the way subsequent points of the shock are estimated is by intersection of local shocks with Mach waves. For the first point, this Mach wave should lie on the sonic line for the procedure to be correct. In the case of the cylinder, Billig's hyperbola provides an approximation for the whole shock, and therefore the position of the sonic line. In general, the first Mach wave and the sonic line position estimated by Billig do not match. The procedure can be improved by multiplying the Mach angle of all Mach waves by a correction factor that matches these two first lines.

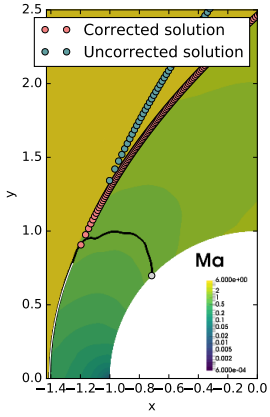

Figure 3: Detachment correction procedure

Fig. 3 illustrates the correction. The CFD sonic line appears in black. The first point of the cylinder geometry with a local tangent wedge that produces a shock with supersonic after shock conditions is shown in grey. From this point, the Mach wave intersecting the first local shock tangent to Billig's solution produces the first point of the uncorrected estimated shock. This Mach wave is far from the estimated sonic line (marked by the end of Billig's solution and the grey cylinder point). Matching this Mach wave with the estimated sonic line gives the correction factor, which is then used for all the other Mach wave computations.

The correction factor is a function of Mach number and is shown in Fig. 4. Applying it to all Mach wave computations greatly improves the results for shock estimation on detached shocks.

In the detached case, the usefulness of the present method is apparent. While shock expansion is not applicable in this situation, tangent wedge still produces useful results. Through the hybridization, the method produces a relatively accurate estimate of shock position, again at the expense of little theoretical foothold.

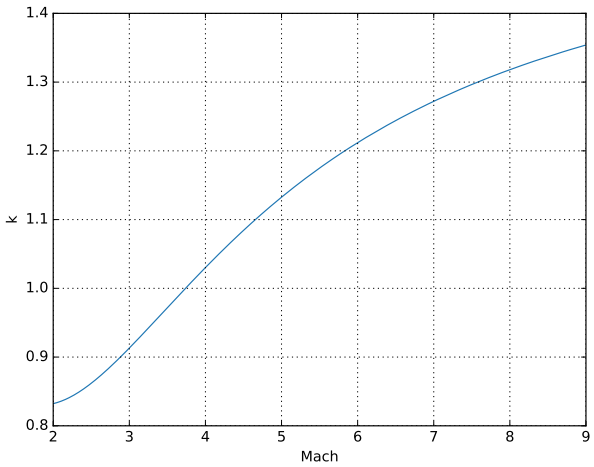

Figure 4: Correction factor for detached shocks

2.1.2. Stream wise body concavities

In previous sections, only convex geometries have been considered. This makes the shock estimation problem simpler, as only one shock is produced by the geometry. When concavities are present, additional shocks are produced. The shocks can now interact with each other or impact the geometry itself. Some modifications have to be introduced to deal with these situations.

First, the geometry is separated into convex sections and concavity zones. This is done by computing the angle between any two connected segments of the geometry. If that angle increases stream wise, a concavity is detected. This concave zone persists until the angle between segments stops increasing. Then, a new convex section starts. From the point of view of the methodology, each convex section will produce a shock, with each concave zone in between producing a compression fan.

The estimation of the shock structure is as follows. The shock produced by the first convex section is estimated as previously described. The first shock point for the next convex section is placed at the apex of a compression fan

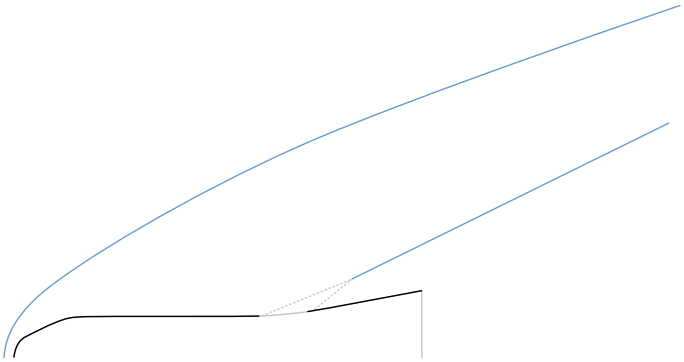

Figure 5: Convex sections and estimated shock

defined in the intervening concave zone. The free stream conditions for these computations are taken as the area weighted mean of the conditions after the first shock, discarding subsonic values. Specifically, the after shock conditions for each shock section are computed using the oblique shock relations. A single value of for the whole shock is then obtained by averaging the results for all shock sections (that produce supersonic values), taking into account their length. While this assumption lacks a strong theoretical justification, it does produce reasonable results. The shock estimation then continues in the usual manner on the next convex section. Therefore, the compression fan is only used to locate the start of the second shock, while its structure is given by the preceding convex section.

Fig. 6 shows the estimation of the shock structure for the HB-2 geometry over CFD. The HB-2[14] is a calibration model defined by the Advisory Group for Aeronautical Research and Development (AGARD) that has been used extensively to study hypersonic flows. The computation captures both the first and the weaker second shock.

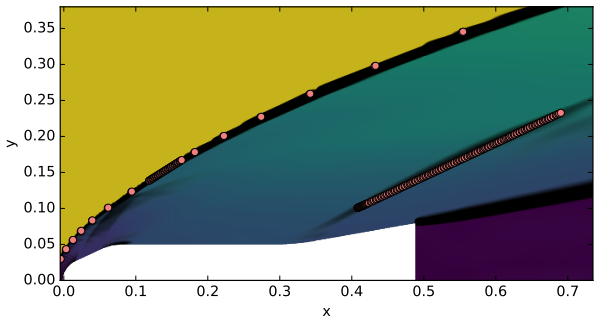

Figure 6: 2D section of HB-2 body. Shock estimation over CFD Mach field with Mach gradient overlay

2.2. Shock estimation in three dimensions

The shock estimation procedure, as explained before, can be applied to three dimensional geometries by carrying out the computation over surface streamlines on the body and then combining these 2D shock estimations into a 3D envelope. While this explanation is straightforward, the application of the method to three dimensional geometries requires some adaptations and introduces certain complications. The geometrical manipulations that were trivial in two dimensions are complex and cumbersome in 3D. For this reason, the implementation of the shock estimation in three dimensions is supported by CGAL[15], a computational geometry library that provides algorithms for common geometrical manipulation tasks.

2.2.1. In-code geometry representation

The geometry in the 2D cases was described by a polyline. In 3D, the geometry is discretized as a triangular surface mesh. The actual implementation of this mesh in code is carried out using a half-edge data structure. While this form of storage is more complex than the common face-vertex approach, it makes for more efficient manipulation and traversal.

2.2.2. 2D sections of 3D geometries

In three dimensions, the procedure presented for 2D is applied over surface streamlines. Two computational geometry algorithms are used to simplify their computation, namely AABB trees and K-D trees.

The procedure followed here to compute surface streamlines is similar to that of the original implementation of HyFlow[16]. An outline of the surface streamline computation is as follows. The free stream velocity vector is projected over the geometry, this creates an interpolation of surface velocities. From the centroid of some facets, the streamlines are integrated backwards, until an stagnation point is reached. The algorithm for selecting streamline origins, the method for surface interpolation of the velocity and the algorithm used to maintain the streamline integration on the surface, differ from HyFlow's original implementation.

For creating the interpolating surface, the free stream velocity vector is projected over each face as

$$\mathbf{u}_s = \mathbf{u}_\infty - (\mathbf{u}_\infty \cdot \mathbf{n}) \cdot \mathbf{n} \tag{6}$$

where $\mathbf{u}_s$ is the surface velocity vector, $\mathbf{u}_\infty$ is the free stream velocity vector, and $\mathbf{n}$ is the normal to the face. From here, the velocity vector associated to each vertex of the geometry is computed as an area averaged mean of the values of the connected faces. The value of the velocity at any point of the surface can now be calculated as the barycentric mean of the values of the vertices of the face to which the point belongs.

The goal of the computation is to uniformly cover the surface with streamlines with the least amount of calculations. The initial point of the inverted streamlines are the centroids of the facets and its selection plays a big role in this reduction of computational cost. The facet centroids are added to a list.

The list is ordered in such a way that elements closer to the beginning of the list are farther with respect to direction of the free stream. That way, the inverted streamlines produced by the first elements will hopefully be the longest ones present over the geometry. The first element of the list is taken as a source. The streamline is integrated, with a time step equal to the quotient between the mean minimum triangle edge length and the maximum surface velocity. At each time step, the computed streamline point is projected back to the surface. This projection requires the computation of the closest triangle and the closest surface point to the computed streamline point. In a naive approach, this would require testing all triangles of the geometry against the point. This approach is infeasible due to the huge computational cost. The projection is simplified by the use of an AABB tree. When the distance between two consecutive computed streamline points fall bellow a certain threshold, a stagnation point has been reached and the integration stops.

Once the streamline is calculated, the centroid list is filtered against the computed streamline. As shown in Fig. 7, the midpoints between streamline points are calculated and spheres are defined with centers on those midpoints and in contact with each other. If a centroid on the list lies inside one of those spheres, it is deleted as a streamline source as it is too close to the current computed streamline. For the first point of the streamline, the sphere is centred on the point instead of on the first segment midpoint. This is done as to filter out the stagnation region together. Again, this nearest neighbors calculation would be highly computationally expensive if preformed in a naive way. The calculation is supported by a K-D tree. After this filtering, the next centroid from the list is chosen as a source and the integration starts again. When the centroid list is empty, the streamline computation ends.

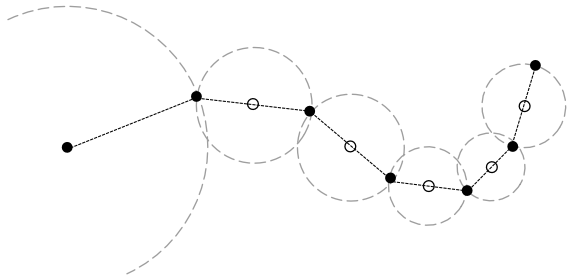

Figure 7: Source filtering

2.2.3. Local inclination in three dimensions

While in 2D the local shocks were computed using tangent wedge method, in 3D the local inclination method that best describes the flow around a point of the geometry depends on the cross-flow curvature of the surface at that point. For mainly flat surfaces, tangent wedge would be more adequate while, for (cross-flow) curved surfaces, some version of tangent cone would better represent the flow. In general, the flow field produced by an elliptic cone at zero angle of attack could be used to address any of these situations.

This local inclination method is conceptually similar to tangent wedge, substituting the flow field used on the estimation from supersonic wedge to supersonic elliptic cone.

First, the eccentricity of the local tangent elliptic cone at each point of the geometry is computed as

$$e = \sqrt{1 - \frac{b^2}{a^2}} \tag{7}$$

where a is the semi-major axis and b the semi-minor axis of the elliptical cross-section of the cone. These values are the maximum and minimum equivalent radii of the local tangent cone in the stream-wise and cross stream-wise directions.

For the stream-wise direction, the radius of the equivalent cone is calculated intersecting the local surface tangent with the previously computed local shock. For the cross stream-wise direction, radius of the equivalent cone is calculated from the curvature of the surface section perpendicular to the axis of the local tangent cone. The curvature of a curve embedded in a surface can be expressed as

$$\overline{k} = k_n \overline{n} + k_g(\overline{n} \times \overline{\tau}) \tag{8}$$

where k_n is the normal curvature, $\overline{n}$ is the normal to the surface at the point, k_g the is the geodesic curvature, and $\overline{\tau}$ is the tangent vector of the curve at the point. The normal curvature can be computed, knowing the principal curvatures of the surface at the point (k_1, k_2), as

$$k_n = k_1 \cos^2\theta + k_2 \sin^2\theta \tag{9}$$

where θ is the angle between the tangent vector of the curve and principal direction corresponding to the maximum principal curvature. From here, the curvature of the curve can be computed multiplying Eq. 8 by $\overline{n}$

$$\overline{k} \cdot \overline{n} = k_n \overline{n} \cdot \overline{n} + k_g(\overline{n} \times \overline{t}) \cdot \overline{n}$$

$$k \cos\psi = k_n$$

where ψ is the angle between the normal to the curve and the normal to the surface. Finally obtaining

$$k = \frac{k_n}{\cos\psi} \tag{10}$$

and $r = 1/k$. As stated before, a and b will be the maximum and minimum of these computed radii. The principal curvatures and principal directions are computed using the "Estimation of Local Differential Properties of Point-Sampled Surfaces" [17] package of CGAL. To compute the shock angle that corresponds to the tangent elliptic cone, the perturbation solution shown in Rasmussen[18] is used.

2.2.4. Shock envelope

Once the shocks are computed over the streamlines, the result is a point cloud without connectivity information. To turn this cloud into a triangular mesh, and advancing front meshing procedure is used. This is implemented in code using the Advancing Front Surface Reconstruction[19] module of CGAL.

2.2.5. Detached shocks

In the two dimensional case, the geometry around the stagnation point was approximated by a circle. The detached shock was then described using Billig's engineering formula for cylinders. In three dimensions, the situation is more complicated. A general stagnation point can be locally described by two principal curvatures, k_1 and k_2, along the principal directions. Billig's engineering formula provides stand off distance and shock radius of curvature for spheres and cylinders in supersonic flow. Between a sphere and a cylinder with the same radius (R), there is a continuum of torii with constant maximum curvature ($k_1 = 1/R$) and descending minimum curvature ($k_2 = [0, 1/R]$). Billig's formula can be generalized to describe those torii and therefore describe the local conditions around general stagnation points in supersonic flow.

For stand off distance and minimum radius of curvature, this generalization is computed by blending Billig's solutions for sphere and cylinder. The blending function should take the value for spheres when the ratio $\lambda = k_1/k_2 = \rho_2/\rho_1 = 1$

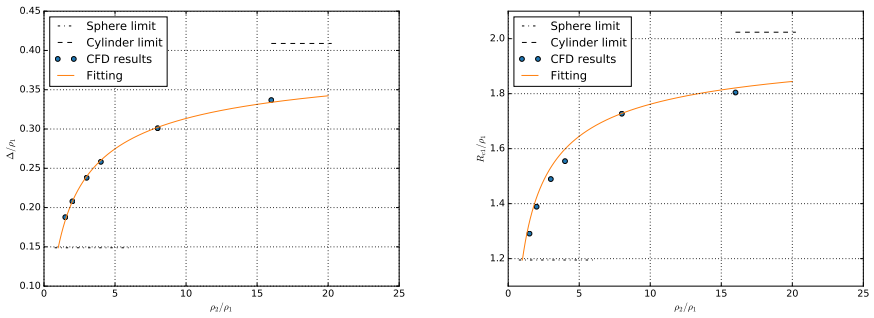

Figure 8: Shock stand off distance as a function of ratio of principal radii of curvature of the geometry ($M = 9$) (left). Minimum radius of curvature of the shock as a function of ratio of principal radii of curvature of the geometry ($M = 9$) (right).

and the value for cylinders when $\lambda = k_1/k_2 = \rho_2/\rho_1 \to \infty$, where ρ are the radii of curvature ($k = 1/\rho$). Inspired by Billig's formulas, the following functional form has been selected

$$f_{tori} = f_{cylinder} \cdot \exp\left(\frac{\ln\left(\frac{f_{sphere}}{f_{cylinder}}\right)}{\lambda^n}\right) \tag{11}$$

where f is either the stand off distance or the radius of curvature of the shock at the vertex in the direction of the maximum curvature. The constant n is fitted to the results of CFD simulations of torii of different ratios of curvatures at Mach number nine. Its value has been computed as $n = 0.57987417201950564$.

Fig. 8 left shows the shock stand off distance divided by the minimum radius of curvature of the geometry at the stagnation point as a function of the ratio of principal radii of curvature of the geometry at the stagnation point. The dots show the results of the CFD simulations, the dashed lines show the limits for sphere and cylinder while the solid line shows the fitted blending function.

Similarly, Fig. 8 right shows the shock minimum radius of curvature at the apex divided by the minimum radius of curvature of the geometry at the stagnation point as a function of the ratio of principal radii of curvature of the

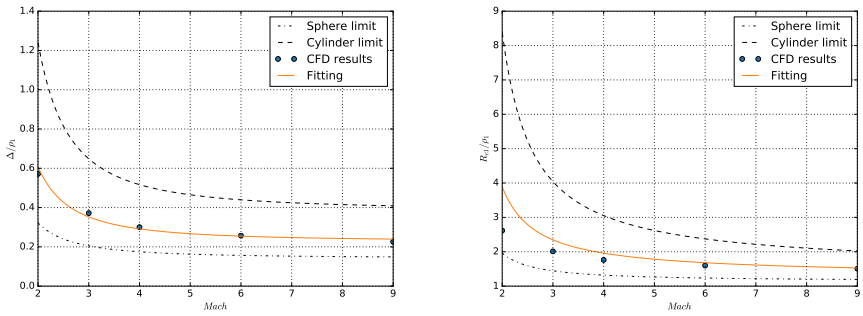

Figure 9: Shock stand off distance as a function of Mach number ($\lambda = 3$) (left). Minimum radius of curvature of the shock as a function of Mach number ($\lambda = 3$) (right).

geometry at the stagnation point. The fitting seems to correctly capture the behavior with variable ratio of radii.

Billig's solution assumes a hyperbola as the shock shape. For extracting stand off distance and shock radius of curvature from the CFD simulations, the equation of the hyperbola has been fitted to the shock position on the CFD simulations for both principal directions. This generates Δ and R_c values which are consistent with the assumptions of Billig's formula.

The functional dependency of the stand off distance with the Mach number is already captured by Billig's formulas for cylinders and spheres. Therefore, the blending function presented in Eq. 11 does not have to contain it explicitly. Fig. 9 left shows this functional dependency for $\lambda = 3$. The fact that the formula shows good agreement with CFD results suggests that the fitting parameter was correctly selected.

For minimum radius of curvature, Fig. 9 right shows the behavior with Mach number. The fitting is slightly worse than in the case of the stand off distance. This might be due to the difficulty of computing the radius of curvature of the shock from the CFD. Finally, Fig. 10 left shows the maximum radius of curvature of the shock as a function of ratio of principal radii of curvature of

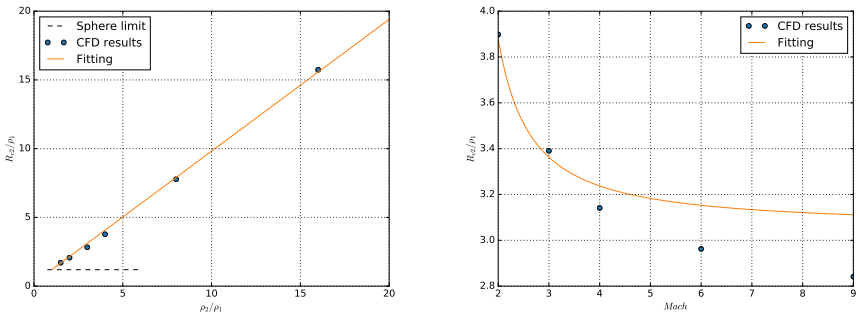

Figure 10: Maximum radius of curvature of the shock as a function of ratio of principal radii of curvature of the geometry ($M = 9$) (left). Maximum radius of curvature of the shock as a function of Mach number ($\lambda = 3$) (right).

the geometry. For this variable, a linear fitting has been selected in the form

$$R_{c2} = n_2\lambda + (R_s - n_2) \tag{12}$$

with $n_2 = 0.9583619656145238$ computed through least squares. The fitting is again performed at Mach number nine. From Fig. 10 right, the fitting for the maximum radius of curvature seems worse than for stand off distance and minimum radius of curvature. The relative error, however, is still below 10%

2.2.6. Concavity in 3D

In three dimensions, two types of concavity are possible, stream wise concavity and cross stream wise concavity. The former refers to the case where the computed streamlines are not convex while the later refers to geometries and flow conditions that present more than one stagnation region. The cross stream direction will be addressed first.

When using the present method on geometries and flow conditions with cross stream wise concavity, even though each streamline is involved in the generation of only one local shock not all local shocks produced over streamlines should coalesce into the same global shock. Indeed, on these geometries, it is expected

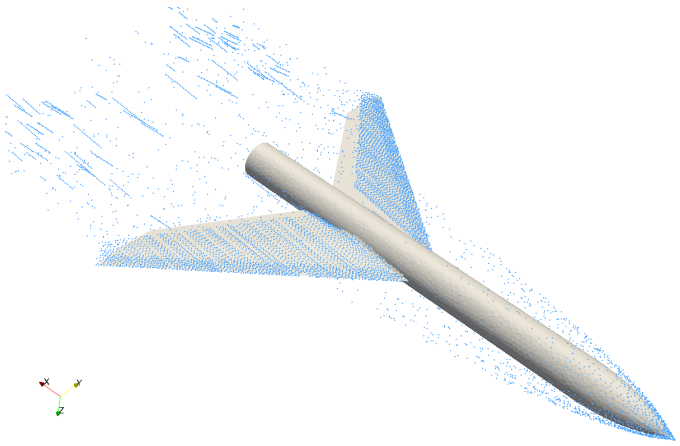

Figure 11: Estimated shock points on a plane-like vehicle

to obtain multiple distinct shocks that interact with each other.

The shock estimation method described in previous sections produces, on a plane-like vehicle, the point cloud shown in Fig. 11. In the current method, all those points would be meshed together to obtain the surface of the global shock. That result is not satisfactory, as the different origin of the shocks is not accounted for. Furthermore, this approach would prevent any meaningful study of shock impingement from the nose shock on the wings or any nose shock - wing shock interaction.

Classifying the local shock point cloud into distinct shocks is infeasible. The solution adopted here is to group together streamlines, applying a clustering algorithm on detected stagnation points. These groups will then be meshed separately to produce distinct shock surfaces.

The algorithm selected to group the stagnation points is the density based spatial clustering of applications with noise (DBSCAN)[20]. As its name suggests, it is a density based clustering algorithm. It has been considered the best for this application as it supports the discovery of clusters of arbitrary shape, it does not require to know the number of clusters in advance, and it has only

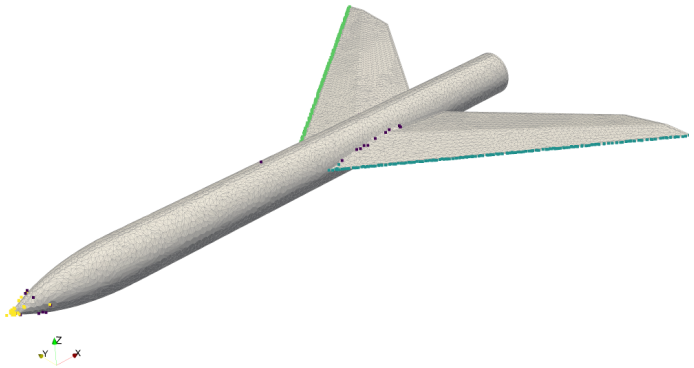

Figure 12: Clusters computed from stagnation points

two control parameters.

Applying DBSCAN on the stagnation points detected on the plane-like vehicle, three clusters are obtained as seen in Fig. 12. The first one is the nose stagnation point, while the other two are stagnation lines on the wing leading edge. The non clustered points are considered noise by the algorithm. These three clusters are now ordered in the stream wise direction, with the goal of estimating which clusters are influenced by others. The assumption here is that all clusters upwind of a given one might influence its shock structure. Now that the streamlines are classified, the local shocks can be meshed into these three global shocks. Once the first global shock (nose) is estimated, the flow conditions after it are computed. The streamlines of the other clusters which stagnation points are inside the convex hull of this first shock are assigned these after shock flow conditions instead of using those of the free stream. The local shock and Mach wave inclinations will then be different than those computed if this interference was not taken into account. The procedure continues cluster by cluster in a stream wise order until all clusters are meshed.

The result of this procedure is shown in Fig. 13. The only influence of

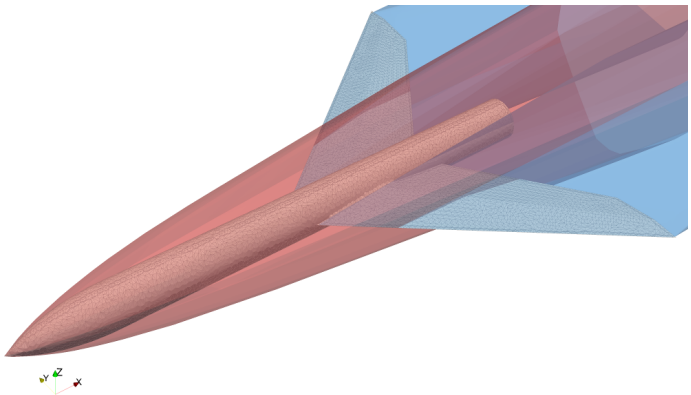

Figure 13: Shock structure around a plane-like vehicle

the nose shock on the wing shock currently taken into account is that the flow conditions inside the former are not free stream. This will influence the shock around the wing roots, which will be less inclined with respect of the free stream as the Mach number used for the computation is lower. The shocks are otherwise considered transparent, with no shock-shock interaction.

Concavity in the stream wise direction is addressed in a way similar to the one employed in 2D. Each streamline would generate more than one shock, as opposed to the cross stream wise concavity case. As such, each streamline is divided into concave and convex sections, with each convex section producing a local shock. Between two of such sections of the same streamline, a compression fan is computed. The start of each convex section is considered a stagnation point for the purpose of clustering.

Once the clusters have been computed, the estimation proceeds as in the cross stream wise concavity case. The clusters are ordered in the stream wise direction and after shock conditions are applied to any stagnation point (meaning here any streamline or streamline section start) contained inside the convex hull of any previously computed shock.

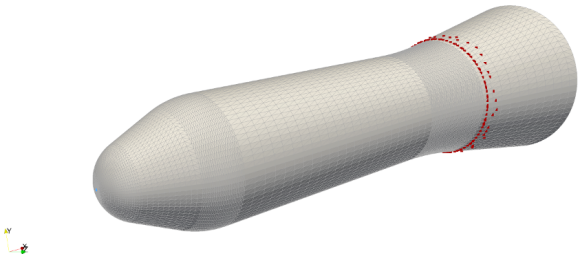

Figure 14: Clusters computed from streamline section starts

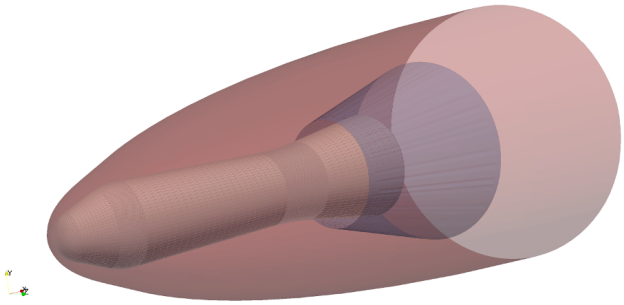

Figure 15: Shock structure around the HB2 vehicle

The procedure is shown, in Figs. 14 and 15, applied to the HB2 vehicle at zero angle of attack. The streamline breaking and clustering produces two distinct clusters, one at the stagnation point on the nose of the vehicle and another annular group at the beginning of the flange.

Two shock estimations are then performed. The bow shock is generated first, using exclusively the first convex part of the streamlines. Then, the conical flange shock is computed over the convex part of the streamlines that lay on the flange. For this zone, the conditions after the bow shock are applied to the computation. It is important to note that this second sock does not start from the surface of the vehicle, but from the convergence point of the compression fan formed on the curved body-flange interface.

3. Verification Cases

The verification of the method has been carried out by comparing the shock estimation against inviscid, steady state CFD simulations. These simulations have been performed in OpenFOAM[21], a continuous mechanics library with precoded solvers, and for the cases considered here, the rhoCentralFoam [22] solver has been adopted. This solver has been used in the past for hypersonic flows[23] and it is considered verified. Therefore, in this work, the results from rhoCentralFoam are taken as reference to which compare the estimated shocks against. All the CFD simulations assume inviscid, steady state flow, using perfect gas model ($\rho = \frac{1}{RT}P$) and run in normalized gas conditions. This means the use of an idealized perfect gas with a molecular weight that results in a speed of sound of $a = \sqrt{\gamma RT} = 1m/s$ for $T = 1K$ and $P = 1Pa$.

To compare the estimated shock with the CFD simulations, the shock position has to be extracted from CFD. There are multiple strategies to do so. Post et al[24] give an overview of the most successful ones. The approach taken in the present work is the one presented by Pagendarm and Seitz [25]. The shock position is described as the locus of the maxima of density gradient in the direction of the velocity. This locus is constructed as a zero level isosurface of the second derivative of the density in the direction of the velocity. To detect only maxima (and discard minima and zones of low gradients) this surface is filtered by selecting only the regions where the first derivative is higher than a certain threshold ϵ. This threshold is case dependent, and is selected as to filter erroneous shocks while still capturing the whole shock structure.

The error in the estimation is computed as the mean minimum distance between the vertices of mesh of the estimated shock and the shock position detected from CFD, non-dimensionalized with the length of the geometry. That is, the minimum distance from each vertex of the mesh of the estimated shock

to the extracted shock surface from CFD is calculated, and the mean of these values is computed.

3.1. Axially-symmetric attached shock. Tangent ogive body

The basic shock estimation procedure, including the effect of the expansion of the flow along a streamline, is illustrated with a tangent ogive body. This geometry is obtained by revolution of a circle arc, which corresponds to the expression

$$r = \sqrt{\rho^2 - (L - x)^2} + R - \rho \qquad x \in [0, L]$$

where

$$\rho = \frac{R^2 + L^2}{2R}$$

being R the maximum ogive radius and L the ogive length.

The shock generated is attached and axially-symmetric. This simple test case is a good benchmark to judge how well the basic method works, where no detachment or three dimensional effects are present.

Fig. 16 shows the estimated shock, compared with CFD, for Mach 9 conditions. As expected, the distance between the estimation and the actual shock increases with the distance to the stagnation point. Fig. 17 left shows the distance between estimated shock and CFD shock, non-dimensionalized with the length of the ogive, along the same section showed in the previous figure. The estimation is quite accurate, with a maximum error below 2.5%. Regarding the behavior with Mach number, the error is expected to decrease as Mach number increases. This comes as a result of the reduced effect of the perturbations reflected from the shock.

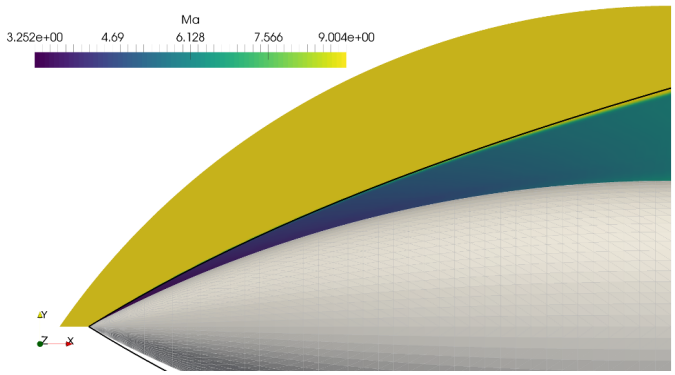

Figure 16: Estimated shock and comparison with CFD for an ogive at Mach 9 and zero angle of attack

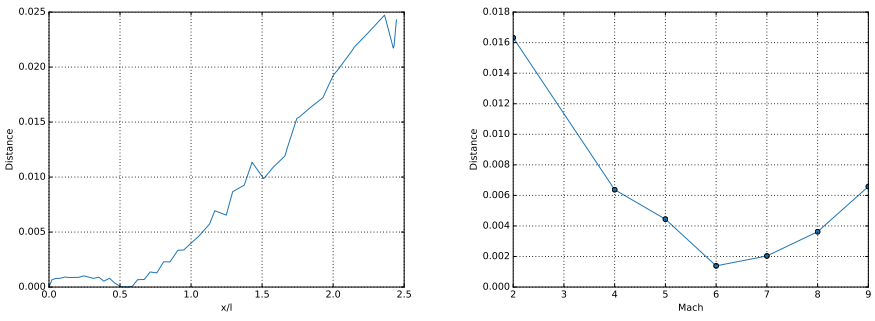

Figure 17: Distance between estimated shock and CFD shock along the surface of the ogive (left). Mean distance between estimated shock and CFD shock as a function of Mach (right).

3.2. Axially-symmetric detached shock. Blunted tangent ogive body

A blunted tangent ogive has been selected to verify the modification for detached shocks. This geometry is obtained by blunting the tangent ogive previously described with a tangent spherical nose. The center of the spherical nose is given by

$$x_0 = L - \sqrt{(\rho - r_n)^2 - (\rho - R)^2}$$

where r_n is the nose radius and R and L are again the maximum ogive radius and the ogive length, being ρ the radius of the supporting circle. From these values, the tangent point is given by

$$y_t = \frac{r_n(\rho - R)}{\rho - r_n}; \qquad x_t = x_0 - \sqrt{r_n^2 - y_t^2}$$

being the apex point

$$x_a = x_0 - r_n$$

Fig. 19 shows the error in the estimated shock with Mach number. The error is higher than in the attached case due to the fact that the estimation is not started from a known condition (attached shock). The solution is started from Billig's generalized formula, which is on its own an approximation. On top of that, the estimation of the principal curvatures of the geometry (triangular mesh) at the stagnation point required by the formula is complex and prone to errors.

3.3. Asymmetrical attached shock. Elliptic tangent ogive body

Next, the effect of non axial symmetry is explored. An elliptic tangent ogive is selected as a test case. This ogive is the same one used as the first test case but scaled in the z direction by a factor of two.

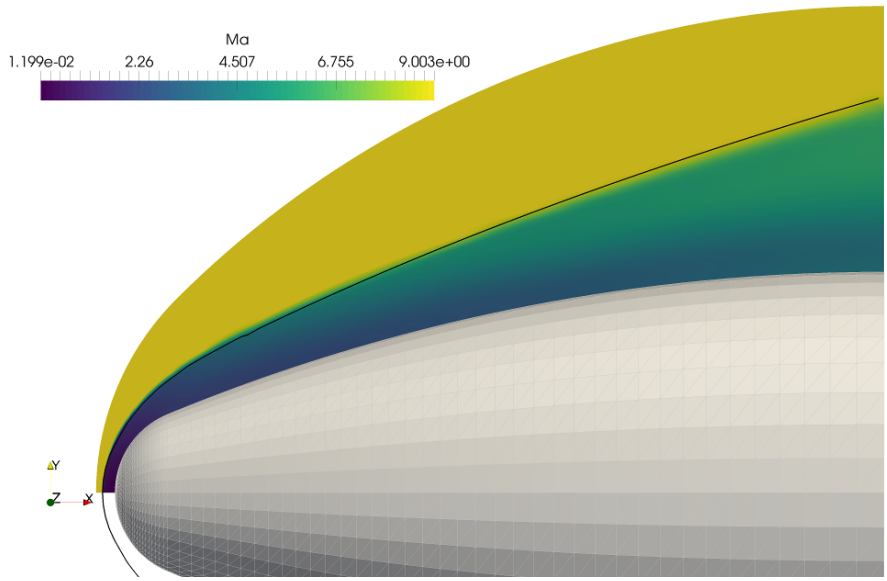

Figure 18: Estimated shock and comparison with CFD for a blunted ogive at Mach 9 and zero angle of attack.

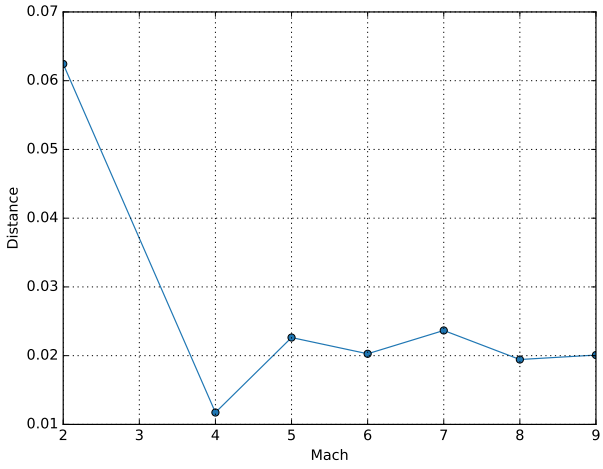

Figure 19: Mean distance between estimated shock and CFD shock as a function of Mach.

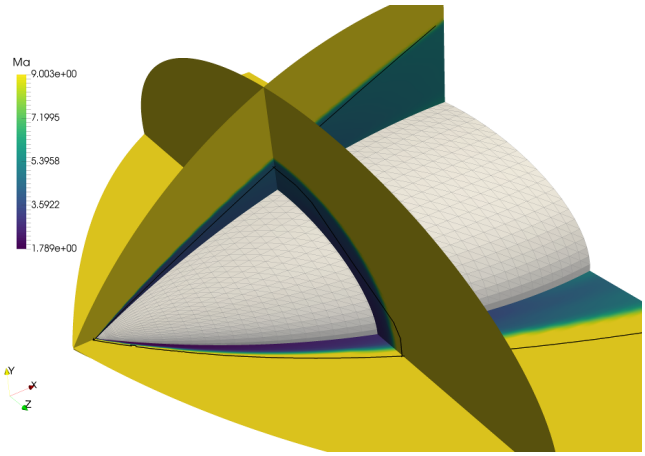

Figure 20: Estimated shock and comparison with CFD for an elliptic ogive at Mach 9 and zero angle of attack.

The lack of axial symmetry suggests the use of elliptic tangent cone theory instead of the traditional (circular) tangent cone theory. Fig. 21 left shows the error in the estimated shock as a function of Mach number. The error is noticeably higher than in the first test case. This is due to the additional approximations being introduced in the elliptic cone perturbation solution, and the need to compute surface curvatures in order to calculate the eccentricity of the local elliptic cone. Even with the required additional approximations, the result obtained using elliptic tangent cone theory is much better than the one obtained using circular tangent cone theory. Fig. 21 right shows a comparison between the two estimations, with the circular cone theory presenting errors twice as high as the ones present with elliptic cone theory.

3.4. Asymmetrical detached shock. Blunted elliptic tangent ogive body

The combined effect of bluntness and lack of axial symmetry provides a challenging test case. The geometry is a blunted elliptic tangent ogive. The main feature of the method tested here is is the generalized Billig formula, required to start the shock estimation. As shown in Fig. 22, the formula accurately captures

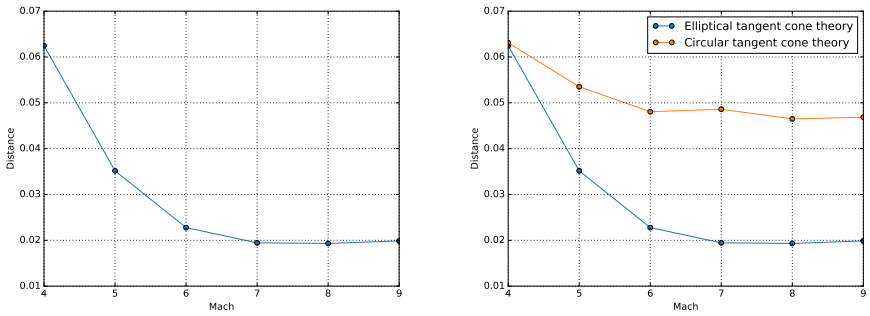

Figure 21: Mean distance between estimated shock and CFD shock as a function of Mach (left). Comparison between estimations using circular tangent cone theory and elliptical tangent cone theory (right).

the shock shape around the stagnation point, with the expected higher error along the principal direction associated with the smallest principal curvature. Fig. 23 shows the error in the estimated shock with Mach number. The error increases with Mach number due to the increased error in the generalized Billig's estimation, as seen in Fig. 10 right. The effects of the curvature calculation are also apparent in this test case.

3.5. Multi-component shock structure. HB-2

The test case selected for analysing the multi-component shock structure estimation is that of HB-2. Here, two shocks are generated, a bow shock around the blunted nose and a conical shock on the aft flange of the geometry. These two shock are shown in Fig. 24 along with the Mach field resulting from a CFD simulation.

The error in the estimated bow shock with Mach number is shown in Fig. 25 left. No clear behavior can be discerned, but the estimation is quite accurate at all tested Mach numbers. While the procedures required for the estimation of the bow shock position have been verified before (blunted tangent ogive), the secondary conical shock showcases new aspects of the computation. The estimation of the position of the secondary shock requires the flow conditions

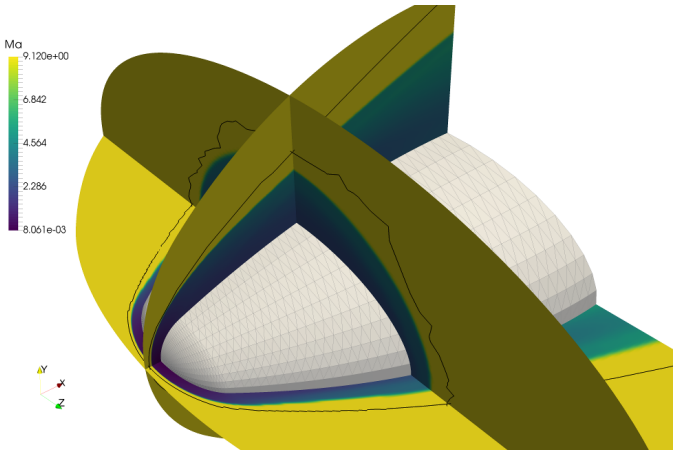

Figure 22: Estimated shock and comparison with CFD for a blunted elliptic ogive at Mach 9 and zero angle of attack

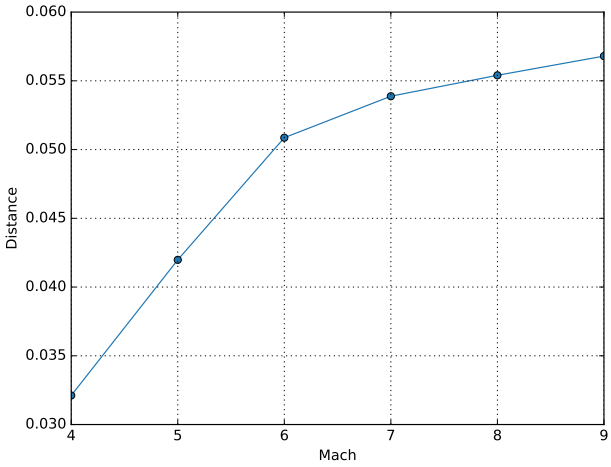

Figure 23: Mean distance between estimated shock and CFD shock as a function of Mach

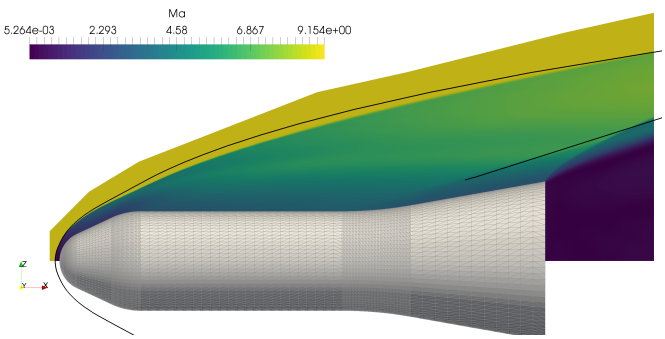

Figure 24: Estimated shock structure and comparison with CFD for the HB-2 at Mach 9 and zero angle of attack

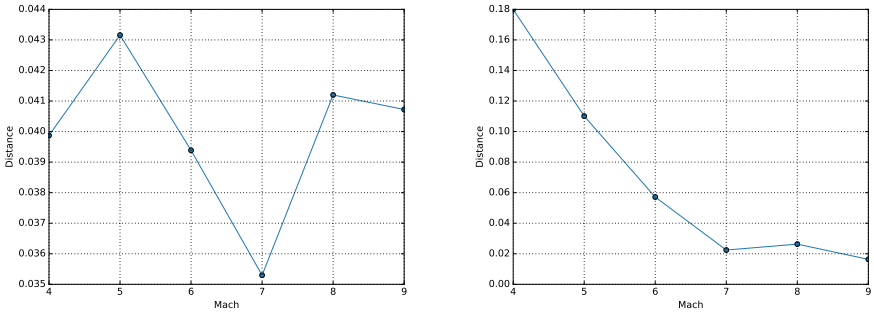

Figure 25: Mean distance between estimated (bow) shock and CFD shock as a function of Mach (left). Mean distance between estimated (flange) shock and CFD shock as a function of Mach (right).

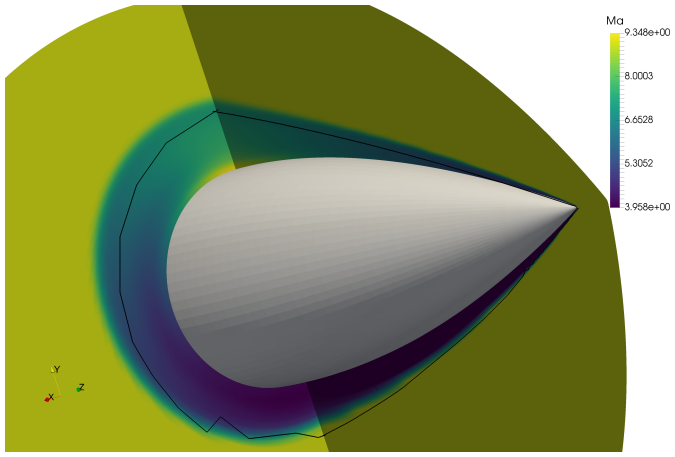

Figure 26: Estimated shock and comparison with CFD for an elliptic ogive at Mach 9 and 10° angle of attack.

after the bow shock, as well as the computation of the compression fan present in the body-flange shoulder. Fig. 25 right shows the error in the estimated secondary shock with Mach number. The geometry of the flange itself is very simple. A lot of aspects contribute to the final error shown in the figure. The error on the estimation of the bow shock propagates to the computation of its after shock flow conditions, that are then used to compute the compression fan that sets the first point of the estimated flange shock and that feed the estimation process. The solution degrades with decreasing Mach number, as it does the estimation of the bow shock and its after shock conditions.

3.6. Angle of attack. Tangent ogive body

In this section, the effect of the angle of attack is studied. The geometry used is the tangent ogive, seen before while verifying attached shocks. The angle of attack introduces an asymmetry in what would otherwise be an axially symmetric flow. As such, elliptical tangent cone might be more suitable than traditional circular cone. Fig. 27 shows the error in the estimated shock while using elliptical tangent cone for the computation of local shocks.

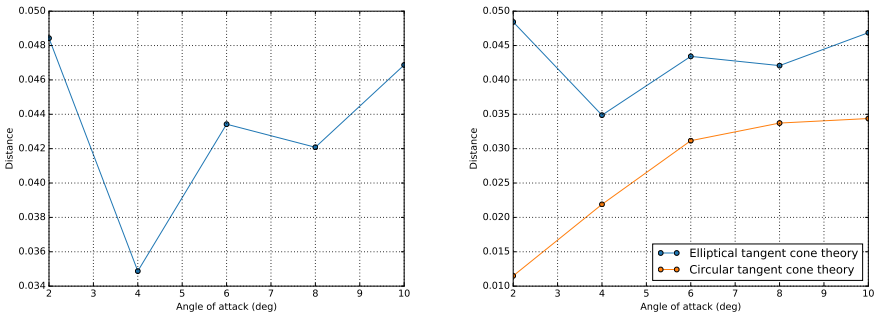

Figure 27: Mean distance between estimated shock and CFD shock as a function of angle of attack (left). Comparison between estimations using circular tangent cone theory and elliptical tangent cone theory (right).

Meanwhile, Fig. 27 right shows a comparison between the error in the estimated shock while using elliptical tangent cone versus circular tangent cone for the computation of local shocks. While in the circular case the errors are overall smaller, they increase with angle of attack as the asymmetry becomes more pronounced. In the elliptic case, however, the errors seem independent from the angle of attack. Elliptic cone appears therefore theoretically better suited for this case due to this error independence, even if practically it is still convenient to use the circular approximation until the problems with elliptic cone are addressed. The higher error comes from the need to compute curvature values on the geometry.

4. Quantitative Assessment via Error Equidistribution

An inviscid, steady-state two dimensional supersonic wedge will be used as a proof of concept to introduce the procedure used for assessing in a quantitative manner the better quality of the mesh obtained with the proposed approach. Two unstructured meshes, with similar number of cells will be considered. The first mesh has a uniform sizing over the entire domain, while the second one has the cell size linked to the estimated position of the shock. After a CFD solution

is obtained in the two cases, the associated error equidistribution is evaluated and compared with the expectation to observe a better equidistribution in the shock-conforming case. An alternative approach would have been to make a comparison in terms of the number of nodes required with the two meshing approaches to get the same level of error equidistribution. This approach has been put aside for the moment since the idea to be validated is that, given a certain number of nodes/elements, which in turns gives indications on the computational resources needed, the proposed approach will allow generating a mesh which is better than a mesh created without information on the shock positioning and supposedly requiring roughly the same computational effort. In this case, the constraint of having a similar number of nodes/elements between the two meshes implicitly determines the resulting mesh spacing in proximity of the shock wave once the spacing at the farfield and along the body is specified.

The wedge considered for this example has a 20 degree semi-angle and it is immersed in inviscid flow at Mach 5. The solution is an oblique shock with 29.8 degree inclination. Meshing has been performed using the open-source code gmsh[26]. The meshes used in this case are quad dominant, produced by recombining a Delaunay triangular mesh. OpenFOAM, the CFD code used in the present work, is cell-centered, it is therefore the cell number what sets the degrees of freedom. The cell counts of the uniform and the shock-conforming mesh are 2,522 and 2,476 respectively. Fig. 29 shows, side to side, the Mach number on the uniform mesh (left), and on the shock-conforming one (right). From a qualitatively point of view, the solution for the mesh that takes into account the shock position looks markedly better. In order to quantitatively appreciate the difference between the two solutions, an error estimation methodology similar to what is done with mesh adaptation techniques has been adopted. An estimation of the local discretization error is computed and its distribution over

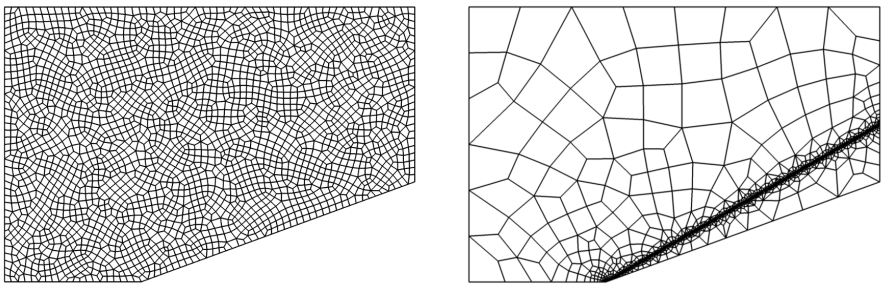

Figure 28: Uniform mesh (left) and shock-conforming mesh (right).

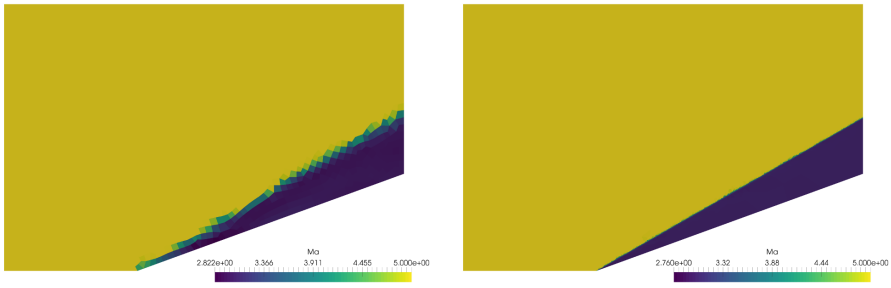

Figure 29: Mach number field for the uniform mesh (left) shock-conforming mesh (right).

the domain is measured.

It has been shown in the literature of mesh adaptation that the discretization error can be associated to the length of the edges of the mesh in a non-Euclidean, namely a Riemannian, space [27]. In this space, the length of the edges are measured by means of a metric tensor that is computed on the basis of the second order spatial derivatives of a suitable flow variable. Error equidistribution is present if the mesh in such a Riemannian space is made by regular polygons, i.e. equilateral triangles / quadrilaterals in 2D or regular tetrahedra / prisms in 3D[3]. Following the approach of Castro-Diaz and collaborators[27], the metric

tensor can be computed as

$$M = R \begin{pmatrix} |\lambda_1| & 0 & 0 \\ 0 & |\lambda_2| & 0 \\ 0 & 0 & |\lambda_3| \end{pmatrix} R^{-1} \tag{13}$$

where λ_i and R are the eigenvalues and a matrix composed from the eigenvectors of the Hessian matrix respectively. The lengths of segments in this metric are then calculated as

$$l = \frac{2}{3} \frac{l_0^2 + l_0 l_1 + l_1^2}{l_0 + l_1} \tag{14}$$

with $l_i = \sqrt{(\gamma')^t M \gamma'}$, where γ' is the derivative of the parametrization of the segment. If the segment is parametrized with s as $x = x_0 + s(x_1 - x_0)$, $\gamma' = x_1 - x_0$ is readily obtained.

In the present analysis, two solutions corresponding to the two meshes are obtained via CFD. The goal is to compare the error distribution of the two solutions on the two meshes. In the present analysis, the Mach number will be adopted as a key variable to compute the Hessian matrix needed for the Riemannian metric tensor. Four mesh-solution pairs are considered, namely (mesh a - solution a), (mesh b - solution a interpolated on mesh b), (mesh a - solution b interpolated on mesh a) and finally (mesh b - solution b). A metric tensor is derived from each one of the four solutions. Then, the lengths of the inter-cell segments (edges of the dual mesh, shown in Fig. 30) on each mesh are computed in this metric and stored on cell faces. Finally, cell values are computed by a surface-weighted sum of cell face values. The error distribution is quantified by the standard deviation of these cell errors. In a vertex centered code, the lengths that would be computed in the Riemannian space would be those of the actual mesh, as it is there where the interpolation error is introduced.

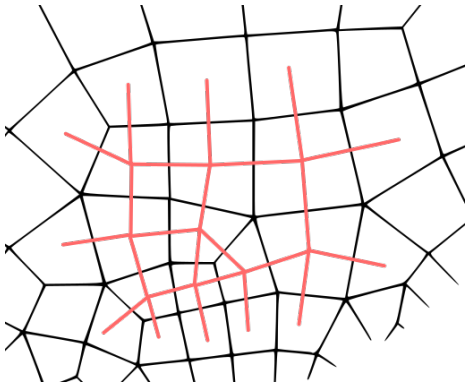

Figure 30: Inter-cell segments (dual mesh) on a cell centered code

	mesh A	mesh B
solution A	0.4435	0.3418
solution B	1.3832	0.6281

Table 1: Standard deviations of error distribution

For a given solution, a smaller standard deviation is expected for the shock refined mesh with respect to the one with uniform sizing. And this is indeed the case, as shown in the table (mesh A is the uniformly sized mesh and solution A is its resulting field). The error is more uniformly distributed (smaller standard deviation) on the shock-conforming mesh (B).

In order to compare the meshes on the same metric, interpolating the solution fields between meshes is required. Therefore an additional interpolation error is introduced. Comparing both meshes on both metrics, which requires A→B interpolation and the reciprocal B→A, mitigates this effect. The need to compare meshes on the same metric comes from the fact that a more refined mesh would produce higher gradients near discontinuities (shocks), therefore a Hessian based error estimator will report higher errors than on a less refined mesh. This can be seen in table, where the mesh A-solution A pair (arguably a bad solution on a bad mesh) shows better error distribution that the mesh B-solution B pair.

5. Shock-conforming Meshes for Supersonic Vehicles

5.1. Sounding Rocket

With the concept demonstrated, the procedure will be applied to relevant engineering cases. The first one is a rocket configuration flying at Mach 6 and zero angle of attack along the vertical direction during the ascent phase. The flow is assumed inviscid and steady-state. The estimated shock structure is fed into the mesh generator. The meshes have been generated as non structured tetrahedral and then dualized into polyhedral meshes to reduce skewness and facilitate the convergence of the flow solver (OpenFOAM in this case). Fig. 31 shows the shock-conforming mesh of 1,505,938 polyhedral cells. The cell sizing takes into account the distance to the shock structure as well as the distance from the nose of the rocket. This is done to address the increasing uncertainty of the shock estimation along the stream-wise direction. It is interesting to observe the presence of the multiple shock waves generated by the rocket body. As indicated earlier, the method, at this time is not capable of addressing the interaction of the shock waves but still the estimated pattern constitutes a good initial approximation of what it is expected to happen in proximity the body.

Figs. 32 and 33 show respectively the solution obtained on the shock-conforming mesh and a comparison with the solution obtained on a uniform mesh with approximately the same number of elements. Overall, the converged the solutions for Mach number on both meshes look markedly similar, with the non-uniform one showing slightly better shock definition. It is in the details, however, where the adapted solution shows its potential. As expected, regions like the nose, the wing-body shock interaction or the domain exit plane are much better resolved in the shock-conforming mesh.

As with the supersonic wedge case, the two meshes are quantitatively compared by measuring the standard deviation of the error distribution. Table 2

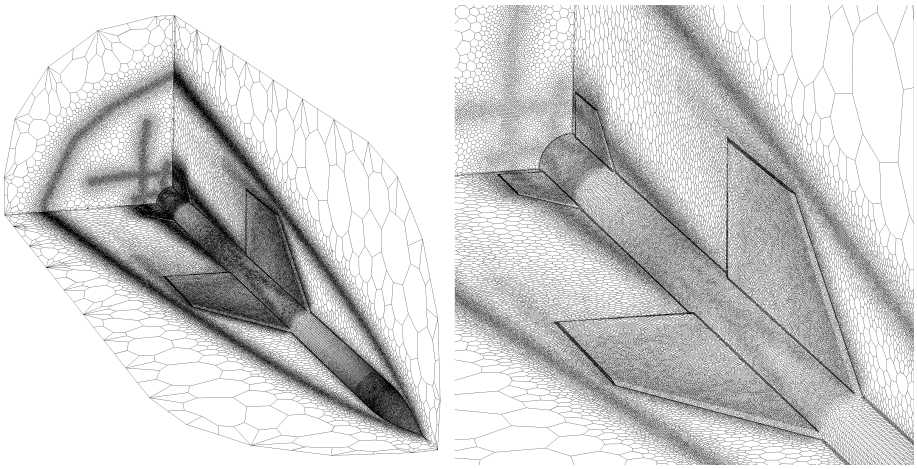

Figure 31: Shock-conforming mesh for the sounding rocket resulting from the estimation of shock waves.

	mesh A	mesh B
solution A	0.7067	0.6383
solution B	0.8988	0.8553

Table 2: Sounding rocket. Standard deviations of error distribution of the two meshes.

shows the standard deviations, where A refers to the uniformly sized mesh and B to the shock-conforming one. The mesh (B) shows a more uniform error distribution, that translates into a mesh closer to optimal.

5.2. X-43 Hypersonic Vehicle

The hypersonic test vehicle X43 is also considered as a pertinent test case given the expected presence of multiple shocks due to its geometrical configuration [28]. A Mach 6 inviscid flow is considered in this case. Differently from the previous case, an unstructured mesh of tetrahedra has been adopted. The flow solver used was the open-source SU2 code from Stanford [29]. Figs. 34 and 35 illustrate the vehicle layout and the shock-conforming mesh. The figures show the presence of the shock waves originating from the leading edge of the vehicle, the vertical and horizontal control surfaces and the thicker refined region

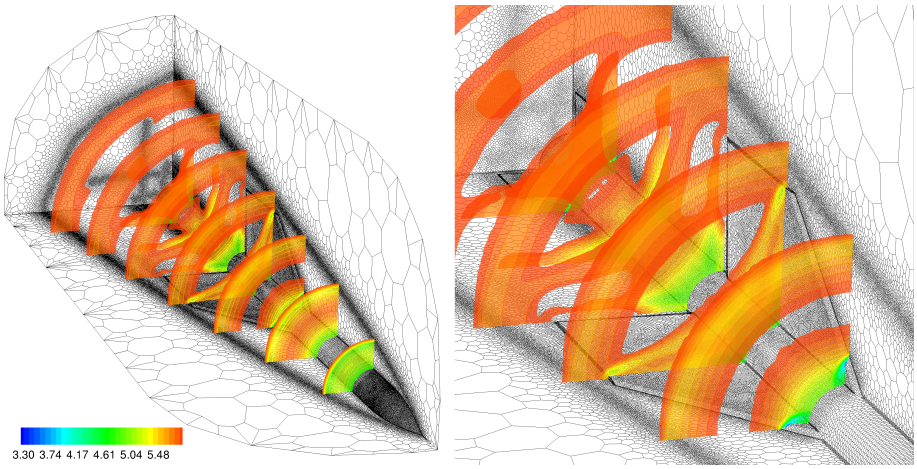

Figure 32: Mach number contours obtained with the shock-conforming mesh.

along the bottom surface due to the presence of the two shocks generated by the double-wedge configuration before the engine intake. The shock-conforming mesh is made by 9,609,782 tetrahedra and 2,515,044 nodes.

Fig. 36 shows the solution comparison with the case of a uniform mesh made by 9,180,203 elements. The visual comparison on the pressure and temperature contours shows the better quality of the solution obtained with the shock-conforming mesh. The right plot, showing temperature contours, evidences how the solution is still noisy and a better, i.e. finer, resolution is still needed to obtain sharp solutions. This could be obtained by triggering a mesh adaptation process that will fine-tune the mesh, but still, considering the given number of nodes/elements and the cost-effectiveness of the proposed approach, the outcome shows a notable improvement with respect to the case of a uniform initial mesh.

A quantitatively analysis based on the Riemannian error estimator has been done to assess the quality of the shock-conforming mesh over the uniform one. Table 3 illustrates the improvement. The same convention as before has been adopted where mesh A and solution A correspond to the uniform case, and mesh

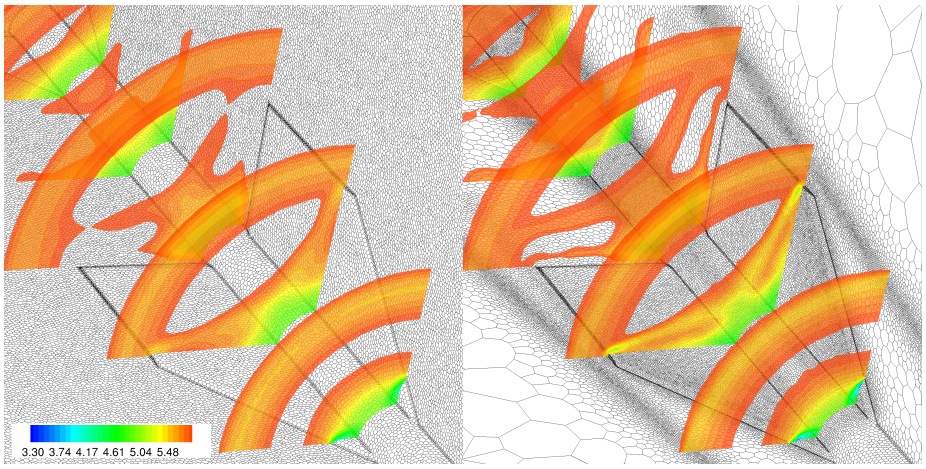

Figure 33: Comparison of Mach number contours. Uniform mesh (left), shock-conforming mesh (right).

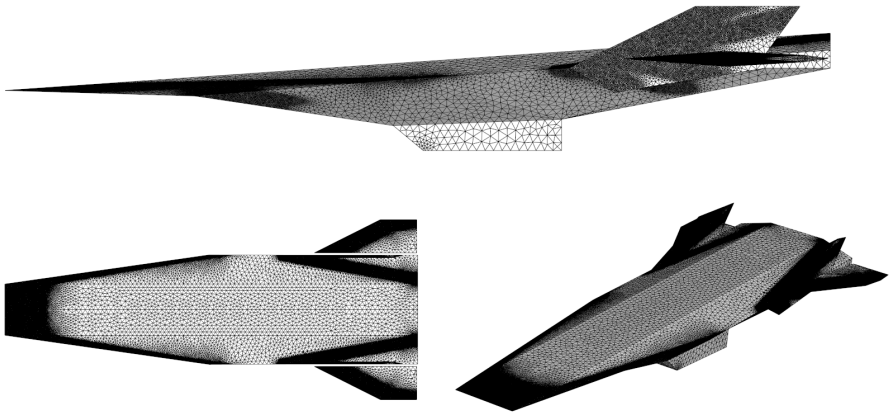

Figure 34: Shock-conforming surface mesh for the X43

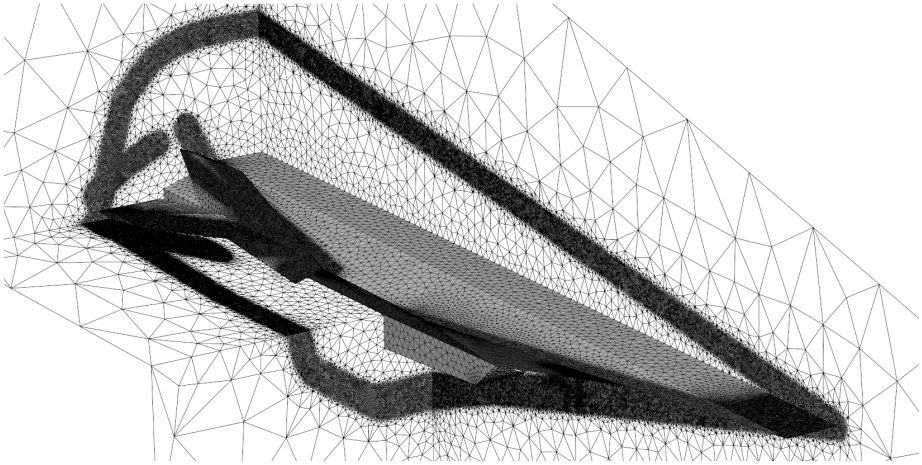

Figure 35: Shock-conforming volume mesh for the X43.

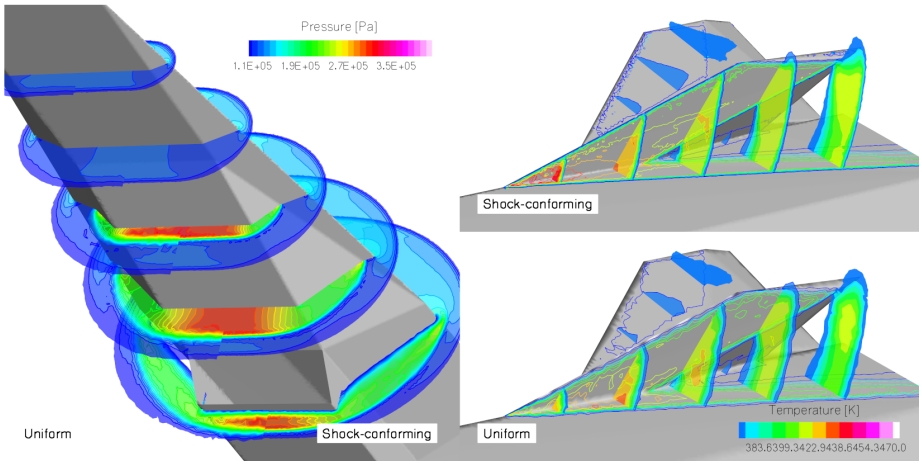

Figure 36: Pressure and temperature contours. Comparison between shock-conforming and uniform mesh.

	mesh A	mesh B
solution A	0.4148	0.3919
solution B	0.5946	0.5023

Table 3: X43. Standard deviations of error distribution for the two meshes.

B and solution B correspond to the shock-conforming case.

6. Final Remarks

A technique for the estimation of the shock wave pattern around supersonic and hypersonic vehicles has been proposed that improves with respect to the existing methods the ability to predict in a very cost effective manner the potentially complex system of waves established around future access-to-space or high-performance airliners. The present formulation revisits existing methods for shock estimation originally conceived for conceptual simple geometries and merges them into a cohesive and optimized approach capable of handling diverse shock wave characteristics (oblique, curved, attached, etc.) in a seamless fashion and in a way that eventually allows for a streamlined interaction with existing open-source mesh generation tools. Despite the fact that the proposed method is limited for the moment to inviscid and steady flow fields and therefore important phenomena like for example shock-boundary layer interaction cannot still be addressed explicitly, the ability to identify regions where shock waves are expected to occur in the vicinity of complex air vehicle could be proven beneficial towards the achievement of grid independent results.

The test cases presented for simple geometries and more complex 3D configurations show the effectiveness of the proposed method. Both qualitative and quantitative analyses have indicated the superior quality of the shock-conformed meshes. Some limitations are recognized for which further efforts are needed. Namely, extending the method to deal with viscous flows and therefore include

the treatment of appropriate boundary layer meshes, improvements in the elliptic cone perturbation method and the surface curvature estimation would be beneficial to obtain even more accurate prediction of shock waves. Another aspect that needs to be taken into further consideration is the formulation of a method to predict in a cost-effective manner the outcomes of the interaction of multiple shocks and potential shock reflection. Eventually, it would be beneficial to introduce a method capable to provide also some preliminary information on the supposedly ideal mesh resolution in proximity of the shock wave. The latter aspect would be relevant especially in the case of viscous flows, where the shock waves are expected to have an internal viscous structure.

Eventually, from an engineering point of view, the merits and the potential of the method can be recognized in allowing for a reduced impact of the user-dependent choice of the initial mesh resolution on the accuracy of the final solution. Alternatively, when combined with mesh adaptation techniques, the method has proven effective in providing a good initial mesh that will allow shortening the long iterative process of subsequent solutions and mesh optimization. Overall the approach is deemed to help reducing the initial uncertainty on CFD-meshing and therefore deemed to promote faster CFD analyses in support of the design of future airliners and access-to-space vehicles.

Acknowledgements

The authors wish to acknowledge the support provided by MBDA UK Ltd and Innovate UK: the work reported herein was undertaken as part of GHandI (TSB 101372), a UK Aerospace Technology Institute project.

References

[1] T. Eggers, Numerical investigation on the potential of steam cooling for the skylon spaceplane in hypersonic flow, in: 28th International Congress of the Aeronautical Sciences (ICAS), June, 2012.

[2] S. J. Laurence, R. Deiterding, G. Hornung, Proximal bodies in hypersonic flow, J. Fluid Mech. 590 (2007) 209–237.

[3] E. F. D'Azevedo, R. B. Simpson, On optimal triangular meshes for minimizing the gradient error, Numer. Math. 59 (1) (1991) 321–348.

[4] S. Pirozzoli, Numerical methods for high-speed flows, Annu. Rev. Fluid Mech. 43 (2011) 163–194.

[5] J. D. Martel, B. Jolly, Analytical shock standoff and shape prediction with validation for blunt face cylinder, in: AIAA Atmospheric Flight Mechanics Conference, January, 2015.

[6] P. Catalano, M. Marini, A. Nicoli, A. Pizzicaroli, CFD contribution to the aerodynamic data set of the vega launcher, J. Spacecr. Rockets 44 (1) (2007) 42–51.

[7] F. S. Billig, Shock-wave shapes around spherical-and cylindrical-nosed bodies., J. Spacecr. Rockets 4 (6) (1967) 822–823.

[8] M. Carter, K. Deere, Grid sourcing and adaptation study using unstructured grids for supersonic boom prediction, in: 26th AIAA Applied Aerodynamics Conference, June, 2008.

[9] E. Gauci, A. Belme, A. Loseille, F. Alauzet, A. Dervieux, A priori error-based mesh adaptation for CFD, in: 19th International Conference on Finite Elements in Flow Problems, April, 2017.

[10] L. Remaki, W. G. Habashi, Toward an Optimal Initial Grid for CFD, in: 43rd AIAA Aerospace Sciences Meeting and Exhibit, January, 2005.

[11] W. Hayes, Hypersonic flow theory, Elsevier, 2012.

[12] A. E. Gentry, D. N. Smyth, W. R. Oliver, The Mark IV Supersonic-Hypersonic Arbitrary-Body Program. Volume II. Program Formulation, Tech. Rep. AFFDL-TR-73-159, Vol. II, Douglas Aircraft Company (1973).

[13] J. D. Anderson, Hypersonic and high temperature gas dynamics, AIAA, 2000.

[14] J. D. Gray, Summary report on aerodynamic characteristics of standard models HB-1 and HB-2, Tech. rep., DTIC Document (1964).

[15] The CGAL Project, CGAL User and Reference Manual, 4.8 Edition, CGAL Editorial Board, 2016.

[16] R. Wuilbercq, Multi-disciplinary modelling of future space-access vehicles.

[17] M. Pouget, F. Cazals, Estimation of local differential properties of point-sampled surfaces, in: CGAL User and Reference Manual, 4.8 Edition, CGAL Editorial Board.

[18] M. P. F. Rasmussen, Waverider Configurations Derived from Inclined Circular and Elliptic Cones, J. Spacecr. Rockets 17 (6) (1980) 537–545.

[19] T. K. F. Da, D. Cohen-Steiner, Advancing front surface reconstruction, in: CGAL User and Reference Manual, 4.8 Edition, CGAL Editorial Board.

[20] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise., in: 2nd International Conference on Knowledge Discovery and Data Mining, Vol. 96, 1996, pp. 226–231.

[21] Openfoam, http://www.openfoam.com/, accessed: 13-04-2016.

[22] C. J. Greenshields, H. G. Weller, L. Gasparini, J. M. Reese, Implementation of semi-discrete, non-staggered central schemes in a colocated, polyhedral, finite volume framework, for high-speed viscous flows, Int. J. Numer. Meth. Fluids 63 (2009) 1–21.

[23] T. Scanlon, R. Cassineli Palharini, C. White, D. Espinoza, V. Casseau, Simulations of rarefied and continuum hypersonic flow over re-entry objects, in: 8th European Symposium on Aerothermodynamics for Space Vehicles, June, 2015.

[24] F. H. Post, B. Vrolijk, H. Hauser, R. S. Laramee, H. Doleisch, The state of the art in flow visualisation: Feature extraction and tracking, Computer Graphics Forum 22 (2003) 775–792.

[25] H.-G. Pagendarm, B. Seitz, An algorithm for detection and visualization of discontinuities in scientific data fields applied to flow data with shock waves, Scientific Visualization: Advanced Software Techniques (1993) 161–177.

[26] C. Geuzaine, J.-F. Remacle, Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities, Int. J. Numer. Meth. Eng. 79 (11) (2009) 1309–1331.

[27] M. Castro-Diaz, F. Hecht, B. Mohammadi, O. Pironneau, Anisotropic unstructured mesh adaption for flow simulations, Int. J. Numer. Meth. Fluids 25 (4) (1997) 475–491.

[28] R. T. Voland, L. D. Huebner, C. R. McClinton, X-43a hypersonic vehicle technology development, Acta Astronautica 59 (1) (2006) 181–191.

[29] F. Palacios, J. Alonso, K. Duraisamy, M. Colonno, J. Hicken, A. Aranake, A. Campos, S. Copeland, T. Economon, A. Lonkar, T. Lukaczyk, T. Taylor,

Stanford university unstructured (SU^2): An open-source integrated computational environment for multi-physics simulation and design, in: 51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, January, 2013.